CSCE 513: COMPUTER ARCHITECTURE

Catalog Description:
513—Computer Architecture. (3) (Prereq: CSCE 211, 212) Design methodologies; processor design; computer arithmetic: algorithms for addition, multiplication, floating point arithmetic; microprogrammed control; memory organization; introduction to parallel architectures.

Prerequisite(s) By Topic:
Basic computer organization
Data structures and algorithms

Textbook(s) and Other Required Material:

John L. Hennessey and David A. Patterson, *Computer Architecture: A Quantitative Approach,* 3rd ed., Morgan Kaufman, 2003. [Required]

David A. Patterson and John L. Hennessey, *Computer Organization and Design: The Software/Hardware Interface*, 2nd edition, Morgan Kaufman, 1998. [Recommended]

Computing Platform: Linux/Unix, Windows XP

Course Objectives: {Assessment Methods Shown in Braces}
1. Describe the principles of computer architecture and organization. {tests}
2. Describe the techniques and principles for the development of high performance computer systems. {tests}
3. Describe the details of extant computer architectures. {tests}
4. Quantitatively analyze aspects of computer architecture and draw conclusions about their performance. {homework, projects, tests}

Topics Covered:
1. Design methodology: system modeling, design levels, hierarchical design (3 hours)
2. Processor design: instruction sets, arithmetic operations (5 hours)
3. Microprogrammed control (3 hours)
4. Algorithms for computer arithmetic: addition, fast addition, multiplication, fast multiplication, division, floating point numbers and operations (8 hours)
5. Memory organization (8 hours)
6. I/O organization (4 hours)
7. Multiprocessor and parallel computer architecture (4 hours)
8. RISC architectures (4 hours)
9. Review and examinations (3 hours)

Laboratory Projects and Other Student Work:
Homework assignments, examinations and class projects involving microbenchmarking, application speedup, simulation, or similar topics.

Difference between Undergraduate and Graduate Work:
Graduate students complete more difficult projects than undergraduate students.

Syllabus Flexibility: Moderate. The instructor may select any textbook that covers the topics listed.

Relationship of Course to Program Outcomes:
The contribution of each course objective to meeting the program outcomes is indicated with the following scale: 3 = major contributor, 2 = moderate contributor, 1 = minor contributor. Blank if not related.

	Program Outcomes										
Course Objectives	1. Logic & Math	2. Computing Fundamentals	3. Apply Computing Principles	4. Work on teams	5. Communicate Effectively	6. Liberal arts & Soc. Sciences	7. Basic Science and Lab Procedures	8. Learn New Tools & Processes	9. Employed upon Graduation	10. Application Area	11. Electronics and Digital Sys Design
1. Describe the principles of computer organization.			3						2		1
2. Describe the techniques and principles for the development of high performance computer systems.			3					2	1		1
3. Describe the details of extant computer architectures.			3						1		1
4. Quantitatively analyze aspects of computer architecture and draw conclusions about their performance.		3	2		2			2	2		2

Estimated Computing Category Content (Semester hours):

Area	Core	Advanced	Area	Core	Advanced
Algorithms			Data Structures		
Software Design			Programming Languages		
Computer Architecture		3			

Estimated Information Systems Category Content (Semester hours):
Computer Information Systems majors do not take this course.

Oral and Written Communication: None

Social and Ethical Issues: None

Theoretical Content: None

Analysis and Design:
The emphasis is on an analytical enumeration of computer architecture and how that examination can be used to understand and improve performance.

Collaborative Work: None

Class/Laboratory Schedule:
Lecture: 3 periods of 50 minutes or 2 periods of 75 minutes per week

Course Coordinator: Duncan Buell

Modification and Approval History:
Initial description, April 1999
Revised, December 2000
Revised June 2005 by Caroline M. Eastman based upon course materials from Kirk Cameron